


Sara Ramos González
Cloud, HPC & Bio
Máster en Bioinformática
Universidad de Murcia
Curso académico 2023-24


Resolución apartado 3.2 – Laboratorio 8: GCP y Kubernetes


- Captura del apartado “actividad” de mi perfil donde se muestra la realización del desafío “GSP304 Build and Deploy a Docker Image to a Kubernetes Cluster”:





Sara Ramos Gonzalez
Miembro desde 2024
150 puntos

Tu perfil no es público y no se puede acceder a él. [Hacer público el perfil](#)


Rutas


Actividades


Tabla de clasificación


Insignias

Curso

Lab

Quest

Cuestionario

Tarea calificada por compañeros

Juego

En curso

Completo

Actividad	Tipo	Fecha de inicio	Fecha de finalización	Puntuación	Correcto
Build and Deploy a Docker Image to a Kubernetes Cluster: Challenge Lab	Lab	hace 1 hora	hace 1 minuto	75.0/100.0	


Sin embargo, la puntuación es solo de un 75 sobre 100, y en el siguiente pdf quiero intentar demostrar que he realizado correctamente la tarea 4 correspondiente al 25% final que no me ha reconocido el laboratorio.

- Tarea 4 del desafío:

Tarea 4: Implementa la aplicación en el clúster de Kubernetes

- Si bien la aplicación está configurada para responder a las solicitudes HTTP en el puerto 8000, debes configurar el servicio para que responda a las solicitudes web normales en el puerto 80. Cuando configures el clúster para la aplicación de ejemplo, asigne el nombre `echo-web` a la implementación.

Haz clic en **Revisar mi progreso** para verificar el objetivo.




Comprobar que se haya implementado una aplicación en el clúster

[Revisar mi progreso](#)

Assessment Completed!

Haz clic en **Revisar mi progreso** para verificar el objetivo.



Comprobar que haya un servicio que responda a solicitudes como Echo-app

[Revisar mi progreso](#)

Please make sure your deployed application service is respond correctly.

- Primero, he creado una nueva versión de la imagen echo-app (v3) en la que he cambiado la aplicación para indicar que debe escuchar en el puerto 8080. Después, he creado un servicio con esta imagen (echo-web) y lo he expuesto en dicho puerto:

```
student_02_1939e24b0335@cloudshell:~ (qwiklabs-gcp-02-6278b49ac092) $ kubectl get services
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
echo-web	LoadBalancer	10.32.6.78	104.197.195.55	8080:32721/TCP	36m

La justificación de que funciona:

Hello Node Kubernetes | Google

Crea e implementa una imagen

Detalles de la imagen - Contain

Cloud Shell

104.197.195.55:8080

No es seguro 104.197.195.55:8080

Echo Test
Version: 1.0.0
Hostname: echo-web-78475c696-jd7nt
Host ip-address(es): 10.28.1.12

- Sin embargo, como el laboratorio no reconocía mi progreso, probé a crear otro servicio (echo-webv1) expuesto esta vez en el puerto 8000 y utilizando la imagen original creada al inicio del laboratorio:

```
student_02_1939e24b0335@cloudshell:~ (qwiklabs-gcp-02-6278b49ac092) $ kubectl get services
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
echo-web	LoadBalancer	10.32.6.78	104.197.195.55	8080:32721/TCP	42m
echo-webv1	LoadBalancer	10.32.6.160	34.122.128.211	8000:30405/TCP	26m

La justificación de que también funciona:

```
Echo Test
Version: 1.0.0
Hostname: echo-webv1-6895598c7f-p8kvm
Host ip-address(es): 10.28.1.13
```

Finalmente, el laboratorio no me ha reconocido ninguno de los dos servicios, aunque no termino de comprender si es un error mío o es que el desafío no me lo ha podido reconocer por algún error externo.

* Adjunto también la justificación de las versiones de la imagen creadas. La v1 es la original que se te pide al inicio del ejercicio. En la v2 cambié el fichero de la aplicación main.go para que usase el puerto 8080. Sin embargo, después me di cuenta de que también tenía que cambiar el puerto en el Dockerfile, por lo que creé la v3 en la que ambos ficheros están cambiados:

Container Registry

Imágenes

Configuración

Imágenes

BORRAR

Container Registry está obsoleto. Después del 15 de mayo de 2024, Artifact Registry alojará imágenes para el dominio gcr.io en proyectos sin uso anterior de Container Registry. [Más información](#)

echo-app

gcr.io > quiklabs-gcp-02-6278b49ac092 > echo-app

Filtro

Ingresar el nombre o el valor de la propiedad

<input type="checkbox"/>	Nombre	Etiquetas	Tamaño virtual	Fecha de creación	Subido	Vulnerabilidades
<input type="checkbox"/>	d297c77a2afa	v3	5.2 MB	hace 37 minutos	hace 21 minutos	2
<input type="checkbox"/>	9890993da829	v2	5.2 MB	hace 37 minutos	hace 36 minutos	2
<input type="checkbox"/>	c3a0bbb5f6dd	v1	5.2 MB	hace 1 hora	hace 58 minutos	2

- Fotos del Dockerfile y main.go cambiados para el puerto 8080:

```
student_02_1939e24b0335@cloudshell:~ (qwiklabs-gcp-02-6278b49ac092) $ ls
Dockerfile echo-web.tar main.go manifests README-cloudshell.txt README.md
student_02_1939e24b0335@cloudshell:~ (qwiklabs-gcp-02-6278b49ac092) $ cat Dockerfile
FROM golang:1.8-alpine
ADD . /go/src/echo-app
RUN go install echo-app

FROM alpine:latest
COPY --from=0 /go/bin/echo-app .
ENV PORT 8080
CMD ["/echo-app"]
```

```
student_02_1939e24b0335@cloudshell:~ (qwiklabs-gcp-02-6278b49ac092)$ cat main.go
/**
 * Copyright 2017 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

// [START all]
package main

import (
    "fmt"
    "log"
    "net/http"
    "net"
    "strings"
    "os"
)

func main() {
    // use PORT environment variable, or default to 8000
    port := "8080"
    if fromEnv := os.Getenv("PORT"); fromEnv != "" {
        port = fromEnv
    }
}
```